Package 'mcbette'

August 19, 2024

Title Model Comparison Using 'babette'

Version 1.15.3

Maintainer Richèl J.C. Bilderbeek <rjcbilderbeek@gmail.com>

Description 'BEAST2' (<https://www.beast2.org>) is a widely used Bayesian phylogenetic tool, that uses DNA/RNA/protein data and many model priors to create a posterior of jointly estimated phylogenies and parameters.
'mcbette' allows to do a Bayesian model comparison over some site and clock models,
using 'babette' (<https://github.com/ropensci/babette/>).

License GPL-3

RoxygenNote 7.3.2

VignetteBuilder knitr

URL https://github.com/ropensci/mcbette/

BugReports https://github.com/ropensci/mcbette/issues

Imports babette (>= 2.3), beautier (>= 2.6.2), beastier (>= 2.4.6), curl, devtools, mauricer (>= 2.5), Rmpfr, testit, txtplot

Suggests ape, ggplot2, hunspell, knitr, lintr, markdown, nLTT, phangorn, rappdirs, rmarkdown, spelling, stringr, testthat (>= 2.1.0), tracerer

Language en-US

Encoding UTF-8

SystemRequirements BEAST2 (https://www.beast2.org/)

NeedsCompilation no

Author Richèl J.C. Bilderbeek [aut, cre] (<https://orcid.org/0000-0003-1107-7049>),
Joëlle Barido-Sottani [rev] (Joëlle reviewed the package for rOpenSci, see https://github.com/ropensci/software-review/issues/360),
Vikram Baliga [rev] (Vikram reviewed the package for rOpenSci, see https://github.com/ropensci/software-review/issues/360, <https://orcid.org/0000-0002-9367-8974>)

Repository CRAN

Date/Publication 2024-08-19 08:40:06 UTC

Contents

`calc_weights` *Calculate the weights for each marginal likelihood*

Description

Calculate the weights for each marginal likelihood

Usage

```
calc_weights(marg_liks)
```

Arguments

`marg_liks`	(non-log) marginal likelihood estimates

Value

the weight of each marginal likelihood estimate, which will sum up to 1.0

Author(s)

Richèl J.C. Bilderbeek

Examples

```
# Evidences (aka marginal likelihoods) can be very small
evidences <- c(0.0001, 0.0002, 0.0003, 0.0004)

# Sum will be 1.0
calc_weights(evidences)

beastier::remove_beaustier_folders()
beastier::check_empty_beaustier_folders()
```

---

can_run_mcbette *Can 'mcbette' run?*

Description

Can 'mcbette' run? Will return TRUE if:

- (1) Running on Linux or MacOS
- (2) BEAST2 is installed
- (3) The BEAST2 NS package is installed

Usage

```
can_run_mcbette(beast2_folder = beastier::get_default_beast2_folder())
```

Arguments

beast2_folder	the folder where the BEAST2 is installed. Note that this is not the folder where the BEAST2 executable is installed: the BEAST2 executable is in a subfolder. Use get_default_beast2_folder to get the default BEAST2 folder. Use get_default_beast2_bin_path to get the full path to the default BEAST2 executable. Use get_default_beast2_jar_path to get the full path to the default BEAST2 jar file.

Value

TRUE if 'mcbette' can run.

Author(s)

Richèl J.C. Bilderbeek

Examples

```
can_run_mcbette()

beastier::remove_beaustier_folders()
beastier::check_empty_beaustier_folders()
```

check_beast2_ns_pkg *Checks if the BEAST2 'NS' package is installed.*

Description

Checks if the BEAST2 'NS' package is installed. Will stop if not

Usage

```
check_beast2_ns_pkg(beast2_bin_path = beastier::get_default_beast2_bin_path())
```

Arguments

`beast2_bin_path`
: path to the the BEAST2 binary file

Value

Nothing. The function will stop with an error message if the BEAST2 'NS' package is not installed.

check_marg_liks *Check if the* `marg_liks` *are of the same type as returned by est_marg_liks.*

Description

stop if not.

Usage

```
check_marg_liks(marg_liks)
```

Arguments

`marg_liks`
: a table of (estimated) marginal likelihoods, as, for example, created by est_marg_liks. This data.frame has the following columns:

- `site_model_name`: name of the site model, must be an element of get_site_model_names
- `clock_model_name`: name of the clock model, must be an element of get_clock_model_names
- `tree_prior_name`: name of the tree prior, must be an element of get_tree_prior_names
- `marg_log_lik`: estimated marginal (natural) log likelihood
- `marg_log_lik_sd`: estimated error of `marg_log_lik`
- `weight`: relative model weight, a value from 1.0 (all evidence is in favor of this model combination) to 0.0 (no evidence in favor of this model combination)

- ess: effective sample size of the marginal likelihood estimation

Use get_test_marg_liks to get a test marg_liks. Use is_marg_liks to determine if a marg_liks is valid. Use check_marg_liks to check that a marg_liks is valid.

Value

Nothing. Will stop with an error message if there is a problem with the input.

---

check_mcbette_state *Check if the* mcbette_state *is valid.*

Description

Check if the mcbette_state is valid. Will stop otherwise.

Usage

```
check_mcbette_state(mcbette_state)
```

Arguments

mcbette_state	the mcbette state, which is a list with the following elements: • beast2_installed TRUE if BEAST2 is installed, FALSE otherwise • ns_installed NA if BEAST2 is not installed. TRUE if the BEAST2 NS package is installed FALSE if the BEAST2 NS package is not installed

Value

Nothing. Will stop if the input is invalid.

Author(s)

Richèl J.C. Bilderbeek

default_params_doc *Documentation of general function arguments. This function does nothing. It is intended to inherit function argument documentation.*

Description

Documentation of general function arguments. This function does nothing. It is intended to inherit function argument documentation.

Usage

```
default_params_doc(
  beast2_bin_path,
  beast2_folder,
  beast2_working_dir,
  beast2_options,
  beast2_optionses,
  clock_model,
  clock_models,
  epsilon,
  fasta_filename,
  inference_model,
  inference_models,
  marg_liks,
  mcbette_state,
  mcmc,
  os,
  rng_seed,
  site_model,
  site_models,
  tree_prior,
  tree_priors,
  verbose
)
```

Arguments

beast2_bin_path	path to the the BEAST2 binary file
beast2_folder	the folder where the BEAST2 is installed. Note that this is not the folder where the BEAST2 executable is installed: the BEAST2 executable is in a subfolder. Use get_default_beast2_folder to get the default BEAST2 folder. Use get_default_beast2_bin_path to get the full path to the default BEAST2 executable. Use get_default_beast2_jar_path to get the full path to the default BEAST2 jar file.

`beast2_working_dir`
: folder in which BEAST2 will run and produce intermediate files. By default, this is a temporary folder

`beast2_options`
: a `beast2_options` structure, as can be created by create_mcbette_beast2_options.

`beast2_optionses`
: list of one or more `beast2_options` structures, as can be created by create_mcbette_beast2_options. Use of reduplicated plural to achieve difference with `beast2_options`

`clock_model`
: a clock model, as can be created by create_clock_model

`clock_models`
: a list of one or more clock models, as can be created by create_clock_models

`epsilon`
: measure of relative accuracy. Smaller values result in longer, more precise estimations

`fasta_filename`
: name of the FASTA file

`inference_model`
: an inference model, as can be created by create_inference_model

`inference_models`
: a list of one or more inference models, as can be created by create_inference_model

`marg_liks`
: a table of (estimated) marginal likelihoods, as, for example, created by est_marg_liks. This data.frame has the following columns:
  - `site_model_name`: name of the site model, must be an element of get_site_model_names
  - `clock_model_name`: name of the clock model, must be an element of get_clock_model_names
  - `tree_prior_name`: name of the tree prior, must be an element of get_tree_prior_names
  - `marg_log_lik`: estimated marginal (natural) log likelihood
  - `marg_log_lik_sd`: estimated error of `marg_log_lik`
  - `weight`: relative model weight, a value from 1.0 (all evidence is in favor of this model combination) to 0.0 (no evidence in favor of this model combination)
  - `ess`: effective sample size of the marginal likelihood estimation

  Use get_test_marg_liks to get a test `marg_liks`. Use is_marg_liks to determine if a `marg_liks` is valid. Use check_marg_liks to check that a `marg_liks` is valid.

`mcbette_state`
: the mcbette state, which is a list with the following elements:
  - beast2_installed TRUE if BEAST2 is installed, FALSE otherwise
  - ns_installed NA if BEAST2 is not installed. TRUE if the BEAST2 NS package is installed FALSE if the BEAST2 NS package is not installed

`mcmc`
: an MCMC for the Nested Sampling run, as can be created by create_mcmc_nested_sampling

`os`
: name of the operating system, must be `unix` (Linux, Mac) or `win` (Windows)

`rng_seed`
: a random number generator seed used for the BEAST2 inference

`site_model`
: a site model, as can be created by create_site_model

`site_models`
: a list of one or more site models, as can be created by create_site_models

`tree_prior`
: a tree prior, as can be created by create_tree_prior

`tree_priors`
: a list of one or more tree priors, as can be created by create_tree_priors

`verbose`
: if TRUE show debug output

Note

This is an internal function, so it should be marked with `@noRd`. This is not done, as this will disallow all functions to find the documentation parameters

Author(s)

Richèl J.C. Bilderbeek

`est_marg_lik` *Estimate the marginal likelihood for an inference model.*

Description

Estimate the marginal likelihood for an inference model.

Usage

```
est_marg_lik(
  fasta_filename,
  inference_model = beautier::create_ns_inference_model(),
  beast2_options = beastier::create_mcbette_beast2_options(),
  os = rappdirs::app_dir()$os
)
```

Arguments

`fasta_filename`	name of the FASTA file
`inference_model`	an inference model, as can be created by create_inference_model
`beast2_options`	a `beast2_options` structure, as can be created by create_mcbette_beast2_options.
`os`	name of the operating system, must be `unix` (Linux, Mac) or `win` (Windows)

Value

a list showing the estimated marginal likelihoods (and its estimated error), its items are::

- `marg_log_lik`: estimated marginal (natural) log likelihood
- `marg_log_lik_sd`: estimated error of `marg_log_lik`
- `esses` the Effective Sample Size

Author(s)

Richèl J.C. Bilderbeek

See Also

- can_run_mcbette: see if 'mcbette' can run
- est_marg_liks: estimate multiple marginal likelihoods

Examples

```
if (can_run_mcbette()) {

  # An example FASTA file
  fasta_filename <- system.file("extdata", "simple.fas", package = "mcbette")

  # A testing inference model with inaccurate (thus fast) marginal
  # likelihood estimation
  inference_model <- beautier::create_ns_inference_model()

  # Shorten the run, by doing a short (dirty, unreliable) MCMC
  inference_model$mcmc <- beautier::create_test_ns_mcmc()

  # Setup the options for BEAST2 to be able to call BEAST2 packages
  beast2_options <- beastier::create_mcbette_beast2_options()

  # Estimate the marginal likelihood
  est_marg_lik(
    fasta_filename = fasta_filename,
    inference_model = inference_model,
    beast2_options = beast2_options
  )

  beastier::remove_beaustier_folders()
}
```

`est_marg_liks` *Estimate the marginal likelihoods for one or more inference models*

Description

Estimate the marginal likelihoods (aka evidence) for one or more inference models, based on a single alignment. Also, the marginal likelihoods are compared, resulting in a relative weight for each model, where a relative weight of a model close to `1.0` means that that model is way likelier than the others.

Usage

```
est_marg_liks(
  fasta_filename,
  inference_models = list(beautier::create_inference_model(mcmc =
    beautier::create_ns_mcmc())),
 beast2_optionses = rep(list(beastier::create_mcbette_beast2_options()), times =
```

```
    length(inference_models)),
  verbose = FALSE,
  os = rappdirs::app_dir()$os
)
```

Arguments

`fasta_filename`	name of the FASTA file
`inference_models`	a list of one or more inference models, as can be created by create_inference_model
`beast2_optionses`	list of one or more `beast2_options` structures, as can be created by create_mcbette_beast2_options. Use of reduplicated plural to achieve difference with `beast2_options`
`verbose`	if TRUE show debug output
`os`	name of the operating system, must be `unix` (Linux, Mac) or `win` (Windows)

Details

In the process, multiple (temporary) files are created (where `[x]` denotes the index in a list)

- `beast2_optionses[x]$input_filename` path to the the BEAST2 XML input file
- `beast2_optionses[x]$output_state_filename` path to the BEAST2 XML state file
- `inference_models[x]$mcmc$tracelog$filename` path to the BEAST2 trace file with parameter estimates
- `inference_models[x]$mcmc$treelog$filename` path to the BEAST2 `trees` file with the posterior trees
- `inference_models[x]$mcmc$screenlog$filename` path to the BEAST2 screen output file

These file can be deleted manually by bbt_delete_temp_files, else these will be deleted automatically by the operating system.

Value

a data.frame showing the estimated marginal likelihoods (and its estimated error) per combination of models. Columns are:

- `site_model_name`: name of the site model
- `clock_model_name`: name of the clock model
- `tree_prior_name`: name of the tree prior
- `marg_log_lik`: estimated marginal (natural) log likelihood
- `marg_log_lik_sd`: estimated error of `marg_log_lik`
- `weight`: relative model weight, a value from 1.0 (all evidence is in favor of this model combination) to 0.0 (no evidence in favor of this model combination)
- `ess`: effective sample size of the marginal likelihood estimation

Author(s)

Richèl J.C. Bilderbeek

See Also

- can_run_mcbette: see if 'mcbette' can run
- est_marg_liks: estimate multiple marginal likelihood of a single inference mode

Examples

```
if (can_run_mcbette()) {

  # Use an example FASTA file
  fasta_filename <- system.file("extdata", "simple.fas", package = "mcbette")

  # Create two inference models
  inference_model_1 <- beautier::create_ns_inference_model(
    site_model = beautier::create_jc69_site_model()
  )
  inference_model_2 <- beautier::create_ns_inference_model(
    site_model = beautier::create_hky_site_model()
  )

  # Shorten the run, by doing a short (dirty, unreliable) MCMC
  inference_model_1$mcmc <- beautier::create_test_ns_mcmc()
  inference_model_2$mcmc <- beautier::create_test_ns_mcmc()

  # Combine the inference models
  inference_models <- list(inference_model_1, inference_model_2)

  # Create the BEAST2 options, that will write the output
  # to different (temporary) filanems
  beast2_options_1 <- beastier::create_mcbette_beast2_options()
  beast2_options_2 <- beastier::create_mcbette_beast2_options()

  # Combine the two BEAST2 options sets,
  # use reduplicated plural
  beast2_optionses <- list(beast2_options_1, beast2_options_2)

  # Compare the models
  marg_liks <- est_marg_liks(
    fasta_filename,
    inference_models = inference_models,
    beast2_optionses = beast2_optionses
  )

  # Interpret the results
  interpret_marg_lik_estimates(marg_liks)

  beastier::remove_beaustier_folders()
  beastier::check_empty_beaustier_folders()
}
```

get_mcbette_state *Get the current state of mcbette*

Description

Get the current state of mcbette

Usage

```
get_mcbette_state(beast2_folder = beastier::get_default_beast2_folder())
```

Arguments

beast2_folder	the folder where the BEAST2 is installed. Note that this is not the folder where the BEAST2 executable is installed: the BEAST2 executable is in a subfolder. Use get_default_beast2_folder to get the default BEAST2 folder. Use get_default_beast2_bin_path to get the full path to the default BEAST2 executable. Use get_default_beast2_jar_path to get the full path to the default BEAST2 jar file.

Value

a list with the following elements:

- beast2_installed TRUE if BEAST2 is installed, FALSE otherwise
- ns_installed TRUE if the BEAST2 NS package is installed FALSE if the BEAST2 or the BEAST2 NS package is not installed

Examples

```
get_mcbette_state()

beastier::remove_beaustier_folders()
beastier::check_empty_beaustier_folders()
```

get_test_marg_liks *Get testing* marg_liks

Description

Get testing marg_liks

Usage

```
get_test_marg_liks()
```

Value

A data frame with marginal likelihoods.

Examples

```
get_test_marg_liks()

beastier::remove_beaustier_folders()
beastier::check_empty_beaustier_folders()
```

`interpret_bayes_factor` *Interpret a Bayes factor*

Description

Interpret a Bayes factor, using the interpretation from [1].

Usage

```
interpret_bayes_factor(bayes_factor)
```

Arguments

`bayes_factor`	Bayes factor to be interpreted

Details

- [1] H. Jeffreys (1961). The Theory of Probability (3rd ed.). Oxford. p. 432

Value

a string with the interpretation in English

Author(s)

Richèl J.C. Bilderbeek

Examples

```
interpret_bayes_factor(0.5)

beastier::remove_beaustier_folders()
beastier::check_empty_beaustier_folders()
```

`interpret_marg_lik_estimates` *Interpret the marginal likelihood estimates*

Description

Interpret the marginal likelihood estimates as created by est_marg_liks.

Usage

```
interpret_marg_lik_estimates(marg_liks)
```

Arguments

`marg_liks`	a table of (estimated) marginal likelihoods, as, for example, created by est_marg_liks. This data.frame has the following columns: • `site_model_name`: name of the site model, must be an element of get_site_model_names • `clock_model_name`: name of the clock model, must be an element of get_clock_model_names • `tree_prior_name`: name of the tree prior, must be an element of get_tree_prior_names • `marg_log_lik`: estimated marginal (natural) log likelihood • `marg_log_lik_sd`: estimated error of `marg_log_lik` • `weight`: relative model weight, a value from 1.0 (all evidence is in favor of this model combination) to 0.0 (no evidence in favor of this model combination) • `ess`: effective sample size of the marginal likelihood estimation Use get_test_marg_liks to get a test `marg_liks`. Use is_marg_liks to determine if a `marg_liks` is valid. Use check_marg_liks to check that a `marg_liks` is valid.

Value

Nothing. This function shows the interpretation via message.

Author(s)

Richèl J.C. Bilderbeek

is_marg_liks *Determine if the* marg_liks *is valid*

Description

Determine if the marg_liks is valid

Usage

```
is_marg_liks(marg_liks, verbose = FALSE)
```

Arguments

marg_liks	a table of (estimated) marginal likelihoods, as, for example, created by est_marg_liks. This data.frame has the following columns: • site_model_name: name of the site model, must be an element of get_site_model_names • clock_model_name: name of the clock model, must be an element of get_clock_model_names • tree_prior_name: name of the tree prior, must be an element of get_tree_prior_names • marg_log_lik: estimated marginal (natural) log likelihood • marg_log_lik_sd: estimated error of marg_log_lik • weight: relative model weight, a value from 1.0 (all evidence is in favor of this model combination) to 0.0 (no evidence in favor of this model combination) • ess: effective sample size of the marginal likelihood estimation Use get_test_marg_liks to get a test marg_liks. Use is_marg_liks to determine if a marg_liks is valid. Use check_marg_liks to check that a marg_liks is valid.
verbose	if TRUE show debug output

Value

TRUE if the argument is a valid marg_liks, FALSE otherwise

mcbette_report *Create a mcbette report, to be used when reporting bugs*

Description

Create a mcbette report, to be used when reporting bugs

Usage

```
mcbette_report(beast2_folder = beastier::get_default_beast2_folder())
```

Arguments

beast2_folder	the folder where the BEAST2 is installed. Note that this is not the folder where the BEAST2 executable is installed: the BEAST2 executable is in a subfolder. Use get_default_beast2_folder to get the default BEAST2 folder. Use get_default_beast2_bin_path to get the full path to the default BEAST2 executable. Use get_default_beast2_jar_path to get the full path to the default BEAST2 jar file.

Value

nothing. It is intended that the output (not the return value) is copy-pasted from screen.

Author(s)

Richèl J.C. Bilderbeek

Examples

```
if(beautier::is_on_ci()) {
  mcbette_report()
}
```

---

mcbette_self_test *Performs a minimal mcbette run*

Description

Performs a minimal mcbette run

Usage

```
mcbette_self_test(beast2_folder = beastier::get_default_beast2_folder())
```

Arguments

beast2_folder	the folder where the BEAST2 is installed. Note that this is not the folder where the BEAST2 executable is installed: the BEAST2 executable is in a subfolder. Use get_default_beast2_folder to get the default BEAST2 folder. Use get_default_beast2_bin_path to get the full path to the default BEAST2 executable. Use get_default_beast2_jar_path to get the full path to the default BEAST2 jar file.

Value

Nothing. Will stop is 'mcbette' cannot run on a minimal, standard input.

plot_marg_liks *Plot the* marg_liks

Description

Plot the marg_liks

Usage

```
plot_marg_liks(marg_liks)
```

Arguments

marg_liks	a table of (estimated) marginal likelihoods, as, for example, created by est_marg_liks. This data.frame has the following columns: • site_model_name: name of the site model, must be an element of get_site_model_names • clock_model_name: name of the clock model, must be an element of get_clock_model_names • tree_prior_name: name of the tree prior, must be an element of get_tree_prior_names • marg_log_lik: estimated marginal (natural) log likelihood • marg_log_lik_sd: estimated error of marg_log_lik • weight: relative model weight, a value from 1.0 (all evidence is in favor of this model combination) to 0.0 (no evidence in favor of this model combination) • ess: effective sample size of the marginal likelihood estimation Use get_test_marg_liks to get a test marg_liks. Use is_marg_liks to determine if a marg_liks is valid. Use check_marg_liks to check that a marg_liks is valid.

Value

a ggplot

Examples

```
plot_marg_liks(get_test_marg_liks())

beastier::remove_beaustier_folders()
beastier::check_empty_beaustier_folders()
```

`set_mcbette_state` *Set the mcbette state.*

Description

Set the mcbette state to having BEAST2 installed with or without installing the BEAST2 NS package.

Usage

```
set_mcbette_state(
  mcbette_state,
  beast2_folder = beastier::get_default_beast2_folder(),
  verbose = FALSE
)
```

Arguments

`mcbette_state`	the mcbette state, which is a list with the following elements: • beast2_installed TRUE if BEAST2 is installed, FALSE otherwise • ns_installed NA if BEAST2 is not installed. TRUE if the BEAST2 NS package is installed FALSE if the BEAST2 NS package is not installed
`beast2_folder`	the folder where the BEAST2 is installed. Note that this is not the folder where the BEAST2 executable is installed: the BEAST2 executable is in a subfolder. Use get_default_beast2_folder to get the default BEAST2 folder. Use get_default_beast2_bin_path to get the full path to the default BEAST2 executable. Use get_default_beast2_jar_path to get the full path to the default BEAST2 jar file.
`verbose`	if TRUE show debug output

Value

Nothing.

Note

In newer versions of BEAST2, BEAST2 comes pre-installed with the BEAST2 NS package. For such a version, one cannot install BEAST2 without NS. A warning will be issues if one intends to only install BEAST2 (i.e. without the BEAST2 NS package) and gets the BEAST2 NS package installed as a side effect as well.

Also, installing or uninstalling a BEAST2 package from a BEAST2 installation will affect all installations.

See Also

- Use get_mcbette_state to get the current mcbette state
- Use check_mcbette_state to check the current mcbette state

Index